Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently amended) A data processing system for qualifying events when an interrupt occurs, comprising:

an interrupt unit control ~~mechanism~~ register for indicating an interrupt type selected to be monitored ~~of a selected type~~;

an interrupt unit, responsive to an interrupt occurring during code execution, for determining [[if]] whether the interrupt is [[an]] of the interrupt type selected to be monitored as indicated by the interrupt unit control register ~~of the selected type~~;

a performance monitoring unit; and

one or more hardware counters located within the performance monitoring unit; wherein the one or more hardware counters count ~~the occurrence of~~ events that occur during processing of the interrupt responsive to ~~a determination by~~ the interrupt unit determining that the interrupt is [[an]] of the interrupt type selected to be monitored as indicated by the interrupt unit control register ~~of the selected type~~.

2. (Currently amended) The system of claim 1, wherein the one or more hardware counters count ~~the occurrence of~~ events that occur during a state of the interrupt ~~of the selected type~~.

3. (Currently amended) The system of claim 2, wherein ~~states~~ the state of the interrupt ~~include~~ includes one of interrupt on, interrupt taken and interrupt acknowledged.

4. (Currently amended) The system of claim 1, wherein the one or more hardware counters count multiple types of events ~~are counted~~ that occur during the processing of the interrupt.

5. (Currently amended) The system of claim 1, wherein the one or more hardware counters count ~~the occurrence of~~ events that occur during processing of the interrupt based on ~~according to~~ the interrupt type of the interrupt ~~during which the events occur~~.

6. (Original) The system of claim 1, wherein the events include clock cycles and cache misses.

7. (Currently amended) The system of claim 1, wherein the interrupt is a first interrupt, and further comprising a second interrupt that interrupts the first interrupt, ~~of the selected type, and~~ wherein the one or more hardware counters ~~count events~~ separately count the events that occur during the processing of the first interrupt ~~of the selected type~~ and events that occur during processing of the second interrupt.

8. (Currently amended) A method of executing instructions in a data processing system ~~on an information processing system~~, comprising ~~the steps of~~:

receiving a signal at a microprocessor of the data processing system for invoking an interrupt, wherein the interrupt includes a plurality of states; and

counting at least one event for a selected state of the plurality of states during processing of the interrupt.

9. (Previously presented) The method of claim 8, wherein the step of counting includes counting at least one event for each of the plurality of states during the processing of the interrupt.

10. (Previously presented) The method of claim 8, wherein the plurality of states include interrupt on, interrupt taken and interrupt acknowledged.

11. (Original) The method of claim 8, wherein the at least one event includes clock cycles and cache misses.

12. (Currently amended) The method of claim 8, wherein the step of counting includes counting multiple types of events for the [[same]] selected state during the processing of the interrupt.

13. (Previously presented) The method of claim 8, wherein the step of counting is performed by one or more hardware counters during the processing of the interrupt.

14. (Currently amended) The method of claim 8, wherein the ~~events are~~ at least one event is counted ~~according to the~~ based on an interrupt type of the interrupt during which the at least one event occurs ~~events occur~~.

15. (Currently amended) The method of claim 8, wherein the interrupt is a first interrupt, and further comprising ~~interrupted by~~ a second interrupt that interrupts the first interrupt, and wherein hardware

counters ~~count events~~ separately count the at least one event that [[occur]] occurs during the processing of the first interrupt and at least one event that occurs during processing of the second interrupt.

16. (Currently amended) A computer program product stored in a computer readable medium in a data processing system for processing instructions, the computer program product comprising:

first instructions for receiving a signal at a microprocessor of the data processing system for invoking an interrupt, wherein the interrupt includes a plurality of states; and

second instructions for counting at least one event for a selected state of the plurality of states during processing of the interrupt.

17. (Previously presented) The computer program product of claim 16, wherein the plurality of states include interrupt on, interrupt taken and interrupt acknowledged.

18. (Previously presented) The computer program product of claim 16, wherein the at least one event includes clock cycles and cache misses.

19. (Currently amended) The computer program product of claim 16, wherein the second instructions comprise instructions for counting count multiple types of events for the [[same]] selected state during the processing of the interrupt.

20. (Currently amended) The computer program product of claim 16, wherein the ~~step of~~ counting is performed by one or more hardware counters during the processing of the interrupt.

21. (Currently amended) The computer program product of claim 16, wherein the at least one event is ~~events are~~ counted ~~according to the~~ based on an interrupt type of the interrupt during which the at least one event occurs ~~events occur~~.

22. (Currently amended) The computer program product of claim 16, wherein the interrupt is a first interrupt, and further comprising ~~interrupted by~~ a second interrupt that interrupts the first interrupt, and wherein hardware counters ~~count events~~ separately count the at least one event that [[occur]] occurs during the processing of the first interrupt and at least one event that occurs during processing of the second interrupt.

23. (Currently amended) A computer implemented method for executing instructions stored in a computer readable medium for qualifying events when an interrupt occurs, comprising:

indicating ~~an interrupt of~~ a selected interrupt type to be monitored;

determining, responsive to an interrupt occurring during code execution, [[if]] whether the interrupt is [[an]] of the interrupt type ~~of the~~ selected to be monitored [[type]]; and

responsive to determining that the interrupt is ~~an interrupt~~ of the interrupt type selected [[type]] to be monitored, counting, by one or more hardware counters located within a performance monitoring unit, ~~the occurrence of~~ events occurring during processing of the interrupt.